

Cahlen Brancheau
Hw6

154

26
40

7

```
begin: addi $t0, $Zero, 0    # Set $t0 to be 0
      addi $t1, $Zero, 1    # Set $t1 to be 1
loop:  slt $t2, $a0, $t1    # if a0 < t1, t2 = 1 else t2 = 0
      bne $t2, $Zero, finish # if t2 != 0 branch to finish
      add $t0, $t0, $t1    # t0 ← t0 + t1
      addi $t1, $t1, 2     # t1 ← t1 + 2
      j loop              # branch to loop
finish: add $v0, $t0, $Zero # Set $v0 to value of $t0
```

Sums all the odd numbers up to input of \$a0 ✓

8

```
Loop: add $s3$t0, $s3$t0, $s3$t0 # s3 ← 2i
      add $s3$t0, $s3$t0, $s3$t0 # s3 ← 4i
      add $s4$t0, $s4$t0, $s4$t0 # s4 ← 2j
      add $s4$t0, $s4$t0, $s4$t0 # s4 ← 4j
      add $s2$t0, $s2$t0, $s2$t0 # s2 ← 2h
      add $s2$t0, $s2$t0, $s2$t0 # s2 ← 4h
Loopi: addi $t0, $s3$t0, $s5 # t0 ← address of A[i]
      lw $t1, 0($t0) # t1 ← A[i]
      add $s1, $s1, $t1 # g ← g + A[i]
      addi $s3, $s3, $s4 # i ← i + j
      bne $s3, $s2, Loop # goto Loop if i != h
```

3

9

lw t0, 0(s0)	# t0 ← A[0]	1
addi t1, s0, 40	# t1 ← address of A[9]	2
addi s0, s0, 4	# s0 ← address of A[2]	3
loop: beq s0, t1, done	# goto done if s0 == t1	4
lw t2, 0(s0)	# t2 ← A[s0]	5
blt t2, t0, skip	# goto skip if A[s0] < A[0]	6
addi t0, t2, 0	# A[0] = A[s0]	7
skip: addi s0, s0, 4	# s0 ← address of A[s0+1]	8
j loop	# goto loop	9
done: ---	t2 = A[s0]	

Loop goes until s0 == t1

s0 = base address of A[] t1 set to address of A+10 (40 bytes/10)

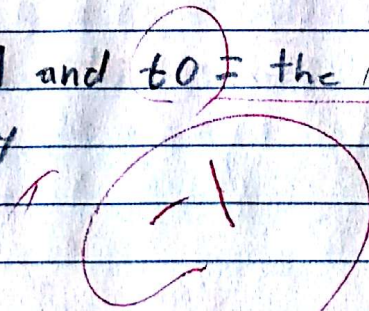
Loop goes through 10 iterations 0-9.

for(int i=0; i<10; i++)

if(A[i] < A[0])

A[0] = A[i];

when $s0 = t1$ $t2 = A[s0]$ and $t0$ is the max value of the first ten elements of the array



10

$\text{add } \$v0, \$zero, \$zero$ # Initialize Count to zero
 loop: $\text{lw } \$v1, 0(\$a0)$ # Load word
 $\text{beq } \$v1, \$zero, skip$ # if word is 0 skip counting
 $\text{addi } \$v0, \$v0, 1$ # Increment count
 skip: $\text{sw } \$v1, 0(\$a1)$ # store word
 $\text{addi } \$a0, \$a0, 4$ # advance to next source
 $\text{addi } \$a1, \$a1, 4$ # advance to next destination
 $\text{bne } \$v1, \$zero, loop$ # Loop if word $\neq 0$

11

a)

$\text{addi } \$t0, \$zero, 1$ # Set $i = 0$
 loop: $\text{beq } \$t0, \text{done}$
 $\text{addi } \$t1, \$t0, 2$ # $t1 \leftarrow 2i$
 $\text{addi } \$t1, \$t1, 4$ # $t1 \leftarrow 4i$
 $\text{addi } \$t2, \$t1, \$a0$ # $t2 \leftarrow \text{address of } a[i]$
 $\text{lw } \$t3, 0(\$t2)$ # $t3 \leftarrow a[i]$ X
 $\text{addi } \$t2, \$t1, \$a1$ # $t2 \leftarrow \text{address of } b[i]$
 $\text{lw } \$t4, 0(\$t2)$ # $t4 \leftarrow b[i]$
 SW $\text{addi } \$t3, \$t4, \$s0$ # $a[i] = b[i] + C$
 $\text{addi } \$t0, \$t0, 1$ # $i = i + 1$
 j loop

done: ---

b) $\text{Prev} \downarrow \text{loop} \downarrow \text{Final check} \downarrow$
 $1 + (10 \times 101) + 1 = 1 + 1010 + 1 = 1012$ ✓

c) $2 \times 101 = 202$

12

a)

$$7 \times 10 + 5 = 75 \quad \checkmark$$

b)

```

add $t1, $s3, $s3 # t1 ← 2i
add $t1, $t1, $t1 # t1 ← 4i
add $t2, $s4, $s4 # t2 ← 2j
add $t2, $t2, $t2 # t2 ← 4j
loop: add $t3, $t1, $s6 # t3 ← address of a[i]
      lw $t0, 0($t3) # t0 ← a[i]
      bne $t0, $s5, Exit # if t0 == s5, branch to Exit
      add $t1, $t1, $t2 # i = i + j
      j loop

```

Exit: ---

$$5 \times 10 + 4 = 54 \quad \times$$

check $k == a[i]$
before adding.

-2

-2